

03/02/00
jc777 U.S. PTO

03-c6-00

PTO/SB/05 (2/98)

Please type a plus sign (+) inside this box → ☐

UTILITY PATENT APPLICATION TRANSMITTAL <small>(Only for new nonprovisional applications under 37 CFR 1.53(b))</small>		Attorney Docket No.	TI-29011	
		First Named Inventor or Application Identifier		Jie Liang
		Title	Image Coding Using Embedded Zerotree Patterns and Bitplanes	
		Express Mail Label No.		EL233151539US

On Page 1 of the specification, before line 1, insert -This application claims priority under 35 USC § 119(e)(1) of provisional application number 60/123,330 filed 03/08/99.--

APPLICATION ELEMENTS <small>See MPEP Chapter 600 concerning utility patent application contents</small>		ADDRESS TO: Assistant Commissioner for Patents Box Patent Application Washington, DC 20231	
1. <input checked="" type="checkbox"/> *Fee Transmittal Form (e.g., PTO/SB/17) <small>(Submit an original, and a duplicate for fee processing)</small>	6. <input type="checkbox"/> Microfiche Computer Program (Appendix)	ACCOMPANYING APPLICATION PARTS 8. <input checked="" type="checkbox"/> Assignment Papers (cover sheet & Documents(s)) 9. <input type="checkbox"/> 37 CFR §3.73(b) Statement (when there is an assignee) <input type="checkbox"/> Power of Attorney 10. <input type="checkbox"/> English Translation Document (if applicable) 11. <input type="checkbox"/> Information Disclosure Statement (IDS)/PTO-1449 <input type="checkbox"/> Copies of IDS Citations 12. <input type="checkbox"/> Preliminary Amendment 13. <input checked="" type="checkbox"/> Return Receipt Postcard (MPEP 503) (Should be specifically itemized) 14. <input type="checkbox"/> *Small Entity Statement(s) <input type="checkbox"/> Statement filed in prior application Status still proper and desired (PTO/SB/09-12) 15. <input type="checkbox"/> Certified Copy of Priority Document(s) if foreign priority is claimed 16. <input type="checkbox"/> Other: <small>*A new statement is required to be entitled to pay small entity fees, except where one has been filed in a prior application and is being relied upon</small>	
2. <input checked="" type="checkbox"/> Specification (preferred arrangement set forth below) - Descriptive title of the Invention - Cross References to Related Applications - Statement Regarding Fed sponsored R&D - Reference to Microfiche Appendix - Background of the Invention - Brief Summary of the Invention - Brief Description of the Drawings (if filed) - Detailed Description - Claim(s) - Abstract of the Disclosure [Total Pages 12]	7. <input type="checkbox"/> Nucleotide and/or Amino Acid Sequence Submission (if applicable, all necessary) a. <input type="checkbox"/> Computer Readable Copy b. <input type="checkbox"/> Paper Copy (identical to computer copy) c. <input type="checkbox"/> Statement verifying identical of above copies		
3. <input checked="" type="checkbox"/> Drawing(s) (35 USC d113) [Total Sheets 2]			
4. Oath or Declaration [Total Pages 2] a. <input checked="" type="checkbox"/> Newly Executed (original or copy) b. <input type="checkbox"/> Copy from a prior application (37 CFR §1.63(d)) (for continuation/divisional with Box 17 completed) [Note Box 5 below] i. <input type="checkbox"/> DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR §1.63(d)(2) and 1.33(b).			
5. <input type="checkbox"/> Incorporation By Reference (useable if Box 4b is checked) The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.			

17. If a CONTINUING APPLICATION, check appropriate box and supply the requisite information below and in a preliminary amendment:

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No: / .

Prior application information: Examiner _____ Group / Art Unit: _____

18. CORRESPONDENCE ADDRESS

☒ Customer Number or Bar Code Label **23494** or ☐ Correspondence address below
(Insert Customer No. or Attach bar code label here)

NAME			
ADDRESS			
CITY	STATE	ZIP CODE	
COUNTRY	TELEPHONE	FAX	

(972) 917-4365 (972) 917-4418

Name (Print/Type)	Carlton H. Hoel	Registration No. (Attorney/Agent)	29,934
Signature		Date	2/28/00

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

INVENTOR INFORMATION

Inventor One Given Name:: Jie
Family Name:: Liang
Postal Address Line One:: 1800 E. Spring Creek Pkwy, Apt. 535
City:: Plano
State or Province:: Texas
Country:: USA
Postal or Zip Code:: 75074
Citizenship Country:: P.R. China

CORRESPONDENCE INFORMATION

Correspondence Customer Number:: 23494
Fax One:: 972-917-4418
Electronic Mail One:: c-hoel@ti.com

APPLICATION INFORMATION

Title Line One:: Image Coding Using Embedded Zerotree Pat
Title Line Two:: terns and Bitplanes
Total Drawing Sheets:: 3
Formal Drawings?:: Yes
Application Type:: Utility
Docket Number:: TI-29011
Secrecy Order in Parent Appl.?:: No

REPRESENTATIVE INFORMATION

Representative Customer Number:: 23494

CONTINUITY INFORMATION

This application is a:: NONPROVISIONAL OF
> Application One:: 60/123,330
Filing Date:: 03-08-1999

Source:: PrintEFS Version 1.0

IMAGE CODING USING EMBEDDED ZEROTREE PATTERNS AND BITPLANES

CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims priority from provisional patent application Serial No. 60/123,330, filed 03/08/99. The following copending application discloses related subject matter: serial no. ...

BACKGROUND OF THE INVENTION

The invention relates to digital image compression and coding and decoding.

For encoding and compressing a single frame as in JPEG or an I frame in MPEG, Shapiro, Embedded Image Coding Using Zerotrees of Wavelet Coefficients, 41 IEEE Tr.Sig,Proc. 3445 (1993) provides a wavelet hierarchical subband decomposition which groups wavelet coefficients at different scales and predicts zero coefficients across scales. This provides a quantization and fully embedded bitstream in the sense that the bitstream of a lower bitrate is embedded in the bitstream of higher bitrates. Figures 2a-2c illustrate a wavelet transform decomposition.

Coding efficiency is, by definition, important to any image compression method. However, increasingly, the features or properties of a coding method play an increasing role in determining the usefulness of the method in practice. Scalability is a very important feature for any image coding method in many applications such as digital photography, Internet, image database, etc. There are many aspects of the scalability property, which can be summarized as follows:

Quality scalability: the bitstream can be parsed (without complicated processing) into sub-bitstream that can be decoded to have different quality reconstruction of the original image. An important case is progressive quality scalability, which means that the bitstream can be decoded progressively, with reconstructed images getting increasingly better with more and more bits received.

Resolution scalability: the bitstream can be parsed into sub-bitstream that represents different resolution of the original image. Progressive resolution scalability is a special case.

Computational complexity scalability: this quality enables the application to adjust the encoding/decoding complexity easily and with minimal change to most of the coding modules.

It is often desirable that both quality and resolution scalability are present in the same bitstream, so that the decoder can choose the desirable resolution and the desirable rate according to its own constraints which can not be decided at the encoding time.

Computational complexity scalability is mostly tied to the memory requirement and speed profile of the method. In many embedded systems, memory is a serious constraint, and speed of encoding/decoding is crucial (for instance, in digital still cameras, shot-to-shot delay is a crucial quality factor in evaluating cameras).

A good coding method should allow a good balance among all these requirements.

SUMMARY OF THE INVENTION

The invention provides a zerotree image coding method which splits the coding symbols into separate value bits and tree bits and codes in children node clusters. This embedded zerotree patterns and bitplanes (EZPAB) method utilizes zerotrees to encode large clusters of zero coefficients and bitplane coding to encode the amplitude of non-zero coefficients.

This has advantages over other wavelet coders (including other zerotree based image coders), in that the method is more scalable, not just in quality and spatial resolutions, but also in computational complexity. EZPAB allows for the optional use of no-entropy coding mode and context-based arithmetic coding mode, which gives a good tradeoff between computational complexity/power consumption and coding efficiency. At the meantime, it consumes minimal amount of memory, which size is independent of image sizes.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows a preferred embodiment coding system.

Figures 2a-2d illustrate wavelet decomposition and tree structure.

Figure 3 is a flow diagram.

Figure 4 illustrates an example.

TI-29011 Page 3

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Overview

The preferred embodiment embedded zerotree pattern and bitplane (EZPAB) coding encodes the wavelet coefficients of an image bitplane-by-bitplane. At each location in the HH, LH, or HL subbands (other than at the finest decomposition level), there are two bits associated with that location in the bitplane: the bit that represents the value of the wavelet coefficient at that location (value-bit), and the bit that represents the significance of the quadtree (zerotree) that is rooted at that location (tree-bit). Encode the value-bit and tree-bit for all four neighbors that have the same parent, and go from one four-coefficient cluster to the next, from the coarsest decomposition level to the finest decomposition level, and from one tree-block to the next. The value-bits can be further classified as significance value-bit, sign value-bit, and refinement (of magnitude) value-bit. Following the scanning procedure, good coding efficiency can be achieved even without any use of entropy coding (that is, simply put the uncoded bits into the bitstream). Another option is to encode these value-bits and tree-bits with context-based arithmetic coding. The coding efficiency is about 10-15% better, and is competitive with the best coder known, with the advantages of less memory requirement and faster speed.

System

Figure 1 illustrates the overall procedure of the EZPAB coder. The image can be input either in block-by-block mode or line-by-line mode. A localized wavelet transform is used to transform the image into wavelet coefficients. Then, the wavelet coefficients are organized into tree structure data, and stored in the tree buffer. Once a zerotree is ready, the system calls on the EZPAB coding module, and feeds the tree structured data to the EZPAB module. When no trees are left in the buffer, it sends a signal to the input module to input more image data (again in either block-by-block or line-by-line fashion). The core part of the coding system is the EZPAB coding module operating on a single full-depth zerotree. The next section shows in more detail the coding procedure for a single zerotree. The bits can be transmitted directly from the EZPAB coding module or can be first arithmetically encoded for transmission. The magnitude scale is determined after all trees are encoded with a

minimum quantization level and in a transmission buffer by simply deleting leading insignificant bitplanes.

EZPAB coding on a single zerotree

Figure 2d shows the structure of a full-depth zerotree for the case of a four-level decomposition. Each non-leaf node (P) has four children nodes. These four-children-node clusters (A,B,C,D) are the basic coding units for EZPAB. Proceed from top level down, and from left to right. For each cluster, first encode their significance or magnitude refinement bits, followed by the tree-bits (except for the finest decomposition level) that indicate the significance of the zerotrees.

Coding of the significance bits $S(A)$, $S(B)$, $S(C)$, and $S(D)$

For each four-children cluster that cannot derive its significance from the coding contexts (that is, they are not part of a zerotree that has already been found insignificant in the current bitplane), explicitly encode their significance.

Encode the significance bits from A to D, and skip any coefficients that have already been found significant. For those coefficients that have been found significant in previous bitplanes, send the refinement (of the magnitude) bits instead. For the newly significant coefficient, encode its sign bit.

The following is the pseudo-code that describes the procedure:

```

for (T in [A, B, C, D])
{
    if (T is already found significant)
        send/encode refinement bit;
        continue;

    if (T is not at the finest decomposition level, or, T is not at location D)
    {
        send/encode  $S(T)$ , the significance bit of T;
        if (T is significant)
            send/encode sign bit of T;
    }
    else /* T is at the finest decomposition level and at location D */
    {
        if (A, B, C) are all insignificant)
            T must be significant, no need to send any significance bit
            send/encode the sign bit of T
        else
            send/encode  $S(T)$ 
            if (T is significant)

```

```

    }
    send/encode the sign bit of T
}

```

Note that nodes D in the finest decomposition level may be skipped when their codes are determined by prior codes, such as when the cluster is significant but nodes A,B, and C are insignificant.

After encoding the value-bits for A,B,C,D, we then encode the four tree-bits that are associated with the four locations. Notice that there is no need to encode tree-bits for the finest decomposition level because a leaf does not have any children. The following describes the procedure for encoding the tree-bits:

```

if (all A,B,C,D tree-bits are insignificant)
    send/encode one bit to indicate that all A,B,C,D are insignificant;
else
{
    for (Tree rooted at [A, B, C, D])
    {
        if (Tree is already found significant)
            continue;

        if (Tree is not at location D)
            send/encode the significance of Tree;
        else
            if (Trees at A,B,C are all insignificant)
                no need to send any bit, Tree at D must be significant
            else
                send/encode the significance bit
    }
}

```

Note that again nodes D may be skipped when their codes are determined by prior codes, such as when the cluster has at least one significant tree bit but nodes A,B, and C have insignificant tree bits.

Also, in all of the above, only encode the tree-bits or value-bits for the four-neighbor clusters that are not part of a larger zerotree. Otherwise, their significance can be derived from the context of the process. Figure 3 is a general flow diagram.

In both procedures, use a "send" or "encode" to indicate which of the two modes is enabled. For a very low-complexity mode, use send, which means that no entropy coding is done here, just put the bits into the bitstream. For a computational intensive mode, use

encode which means a context-based arithmetic coding to encode the bits. For the value-bit, the context is determined by the number of significant coefficients in the same cluster as well as the location of the current coefficient (A,B,C or D), which is a total of $4 \times 4 = 16$ contexts. For the tree-bit, the context is determined by the number of significant trees in the same cluster and the significance of the coefficient at the same location, which is a total of $4 \times 2 = 8$ contexts.

Thus the foregoing provides a coding method that strikes a balance among all the different features. This enables a mode that requires no arithmetic or Huffman coding. The bits that represent zerotree patterns, coefficient signs, and magnitude bits are directly output to the final bitstream. On the other hand, without any change to the coding procedures (the same transform, coefficient scanning, zerotree forming procedures), one can encode those same bits with context-based arithmetic coding. All features and memory requirement remain the same between these two modes, but significant speed up can be achieved with the elimination of arithmetic coding and context forming.

Figure 4 illustrates an example of the coding. In particular, Figure 4 represents the coding of the four nodes in a vertical slice through a full-depth tree; the columns represent the bitplanes with most significant bit to the left and the rows represent the value and tree bits from coarsest to finest decomposition level with the coarsest at the top. Thus the coding proceeds from top to bottom through a tree and proceeds left to right through the bitplanes. At the top of the first column the 0 indicates an insignificant value at the coarsest level in the most significant bitplane. The 1 below this 0 indicates that this coarsest level node is the root of a zerotree (that is, in the most significant bitplane all of the descendant (children) nodes of this full-depth tree are insignificant). Because all children node coefficients are insignificant, they are skipped in the coding as indicated by the remainder of the first column.

The top entry 1 of the second column indicates the coarsest level node is significant in the second most significant bitplane (that is, the value of the corresponding coefficient has its first nonzero bit in this bitplane) and the \pm indicates a second value bit is used to indicate the sign of the coefficient. The 1 in the second row indicates that all of the children are still insignificant and are skipped.

The top entry of the third column indicates the second (refinement) bit of the binary expansion of the coefficient at the coarsest level node. Again, the 1 in the second row indicates a zerotree so the children are again skipped.

The top entry of the fourth column indicates the third (refinement) bit of the coefficient at the coarsest level node. The 0 in the second row indicates that at least one of the children nodes has a significant coefficient in the fourth bitplane. Thus the cluster of four children at the second level have their value bits and then their tree bits encoded. For the second-level child node (of the cluster) in the third row the value bit 0 indicates that the coefficient is still insignificant in the this (fourth) bitplane but the tree bit 0 indicates that at least one descendant in either the third or fourth level is significant. The 1 in the fifth row indicates that this third-level child node coefficient has its leading nonzero bit in this bitplane and the \pm again indicates the sign; the other three nodes in its cluster have their value bits encoded at this time. The 1 in the sixth row indicates that this third-level child is the root of a zerotree, so its four children are skipped. The other three nodes in its cluster have their tree bits encoded at this time.

The fifth column (bitplane) has from top to bottom: the fourth (refinement) bit for the coarsest level node coefficient, the second row skip of the tree bit because the tree was already found significant in the fourth column. The third row 1 and \pm are two bits that indicate this second-level child's coefficient has its first significant (nonzero) bit in this bitplane and its sign. The fourth row tree bit is skipped because the tree was already found significant in the fourth bitplane. The fifth row is the second (refinement) bit for the third-level child's coefficient, and the sixth row 0 indicates that at least one fourth-level child is significant in this bitplane. The seventh row 1 and \pm are two bits that indicate this fourth-level child's coefficient has its first significant (nonzero) bit in this bitplane and its sign. The sixth column (bitplane) shows value bits as refinement bits and skips tree bits because of previous tree significance.

Decoding

The decoding of the bitstream just interprets the bits according to the EZPAB coding to recover the wavelet coefficients (to whatever number of refinement bits usable) and then

performs an inverse wavelet transformation to recover the image. Of course, error correction and other channel coding may be imposed for transmission or storage.

Experimental results

In this section, we provide the coding results using the proposed method. We use both options of the EZPAB coder: option A – no entropy coding, and option B – using arithmetic coding. We choose the commonly used image Lena (512x512). The results include all overhead, including the 4 bytes resynchronization marker for each subband at each bitplane (a significant overhead, but necessary for good error resilience properties).

EZPAB Option A (no entropy coding)		EZPAB Option B (with arithmetic coding)	
Bitrate (bit/pixel)	PSNR (db)	Bitrate (bit/pixel)	PSNR (db)
2.004	45.294	2.193	45.294
0.998	40.337	1.098	40.337
0.478	36.931	0.526	36.931
0.251	34.003	0.284	34.003
0.125	30.882	0.124	30.032

Table 1. Coding Results

CLAIMS

What is claimed is:

1. A method of image coding, comprising:

(a) wavelet transform a portion of a digital image into hierarchical trees of coefficients;
(b) for a first tree of said hierarchical trees and for a first cluster of children nodes

derived from a first parent node of said first tree, after encoding said first parent node
encode a value and tree indication for each of the children nodes in said first cluster wherein

(i) said value and tree indication are skipped when said encoding of said first parent node so
indicates, (ii) said tree indication for a first of said children nodes indicates significance of
descendant nodes of said first of said children nodes, (iii) said value for a first of said
children nodes indicates significance of said first of said children nodes, (iv) said value and
tree indication for one of said children nodes is skipped when determinable from encoding of
the other of said children nodes of said first cluster, and (v) said tree indication is omitted if
said children nodes in said first cluster have no descendant nodes; and

(c) repeat foregoing step (b) for a second, third, ..., and Nth cluster of children nodes
derived from a second, third, ..., and Nth parent node of said first tree; and

(d) repeat foregoing steps (b)-(c) for a second, third, ..., and Mth tree of said
hierarchical trees.

2. The method of claim 1, wherein:

(a) the steps (b)-(c) of claim 1 are repeated for each bitplane of said coefficients.

3. The method of claim 1, wherein:

(a) the steps (b)-(c) of claim 1 include arithmetic coding of said values and tree
indications using a context which for a children node includes other children nodes of a
common cluster.

4. The method of claim 2, wherein:

(a) said value is a member of the group consisting of a significance bit plus a sign bit, an insignificance bit, a refinement bit, and a skip.

5. The method of claim 2, wherein:

(a) said tree indication is a member of the group consisting of a zerotree root, a significant descendant node, and a skip.

ABSTRACT OF THE DISCLOSURE

An image coding method using wavelet decomposition and coding clusters of four-children coefficients bitplane-wise including value-bits for significance and refinement, and tree-bits for tree structure significance.

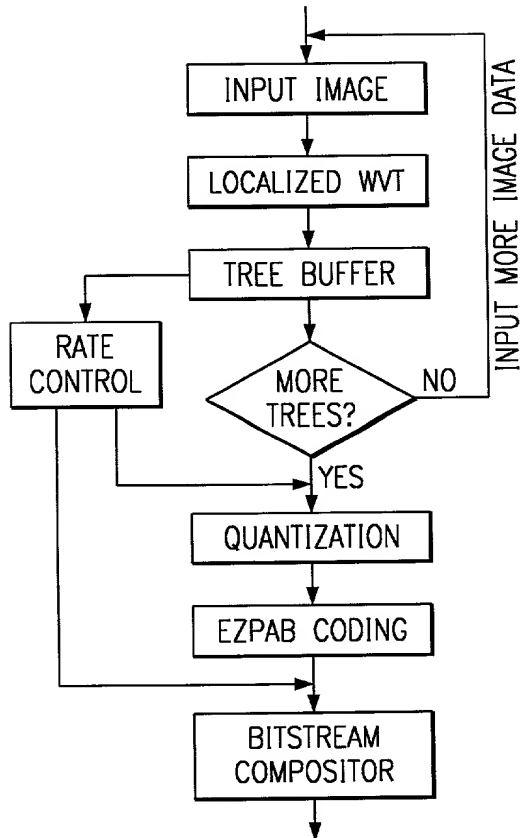


FIG. 1

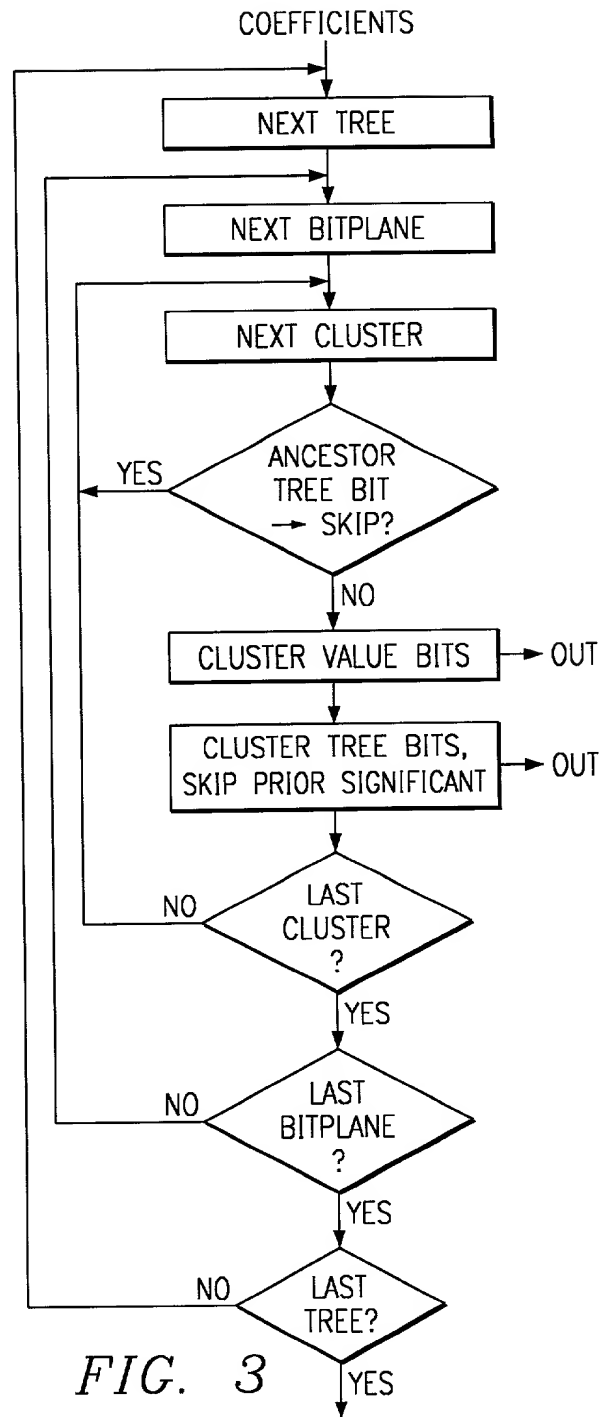


FIG. 3

		MSB	2nd	3rd	4th	5th	6th
COARSEST LEVEL	VALUE	0	1 ±	R	R	R	R
	TREE	1	1	1	0	SKIP	SKIP
SECOND LEVEL	VALUE	SKIP	SKIP	SKIP	0	1 ±	R
	TREE	SKIP	SKIP	SKIP	0	SKIP	SKIP
THIRD LEVEL	VALUE	SKIP	SKIP	SKIP	1 ±	R	R
	TREE	SKIP	SKIP	SKIP	1	0	SKIP
FINEST LEVEL	VALUE	SKIP	SKIP	SKIP	SKIP	1 ±	R

FIG. 4

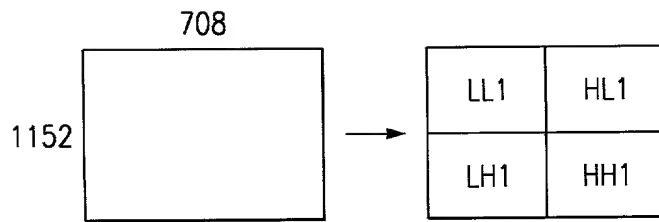


FIG. 2a

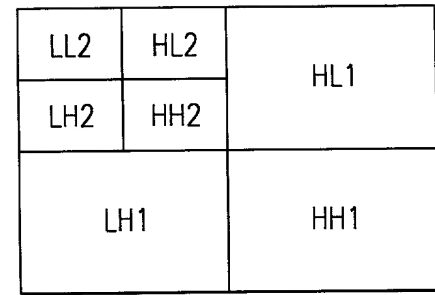


FIG. 2b

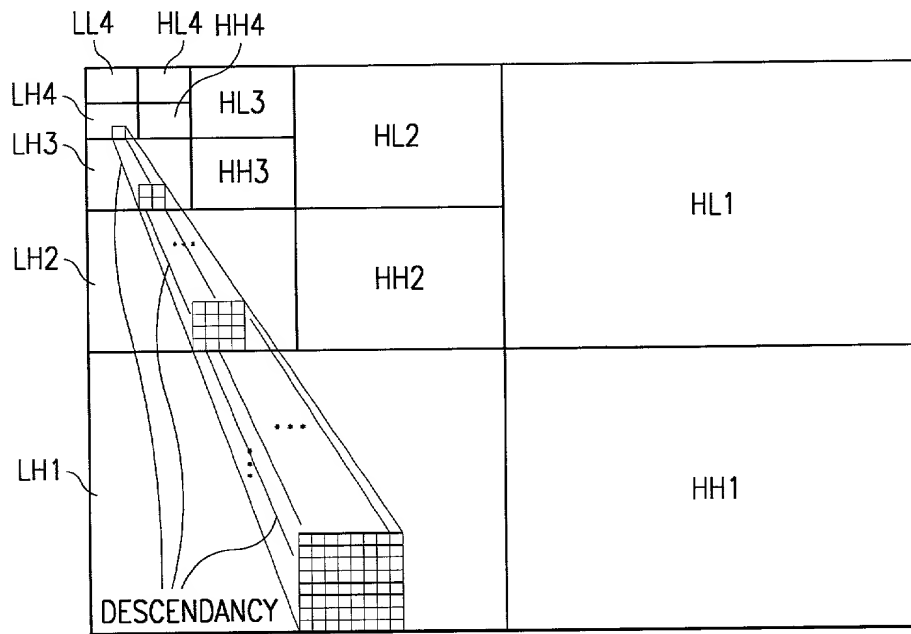


FIG. 2c

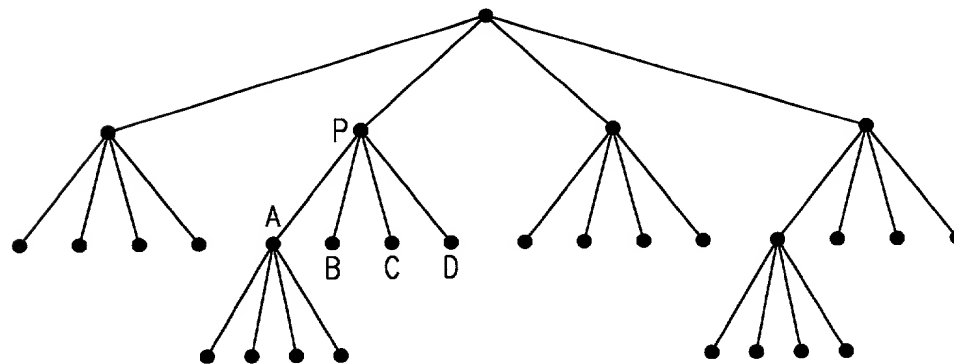


FIG. 2d

COMBINED DECLARATION AND POWER OF ATTORNEY

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name, and I verily believe that I am the original, first and sole inventor if only one name is listed below, or an original, first and joint inventor if plural inventors are named below, of the subject matter which is claimed and for which a patent is sought on the invention entitled: **Image Coding Using Embedded Zerotree**

Patterns and Bitplanes, which:

(a) X is attached hereto.

(b) _____ was filed on _____ as Serial No. _____.

(c) _____ was described and claimed in PCT International Application No.

_____ filed on _____ and as amended under PCT Article 19 on _____ (if any).

I have reviewed and understand the contents of the above identified specification, including the claims as amended by any amendment specifically referred to in the oath or declaration;

No application for patent or inventor's certificate on this invention has been filed by me or by my legal representatives or assigns in any country foreign to the United States of America;

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations Section 1.56(a);

I hereby claim that benefit under Title 35, United States Code Section 120 of any United States applications listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, Section 1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

Serial No. 60/123,330, filed on 03/08/99.

As a named inventor, I hereby appoint the following attorneys to prosecute this application and transact all business in the Patent and Trademark Office connected therewith:

Carlton H. Hoel, Reg. No. 29,934
Richard L. Donaldson, Reg. No. 25,673;
Jay M. Cantor, Reg. No. 19,906
William B. Kempler, Reg. No. 28,228;

Send correspondence and direct telephone calls to:

Carlton H. Hoel
Texas Instruments Incorporated
PO Box 655474, M/S 3999
Dallas, Texas 75265
(972) 917-4365

I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both under Section 1001 of Title 18 of the United States Code, and that such willful statements may jeopardize the validity of the application or any patent issuing thereon.

Name of Inventor: Jie Liang
Residence and Post Office Address: 1800 E. Spring Creek Pkwy., Apt. 535, Plano, Texas 75074
Country of Citizenship: P.R. China

Inventor's signature:

x Jie Liang

Date: x 2/24/2000